COMP 405/505

Advanced Topics in Object-Oriented Design

Comp405/505 Syllabus

COMP 405/505 is a topics-driven exploration of cutting-edge object oriented design issues and concepts including mutable recursive data frameworks, design patterns for lazy evaluation, sorting, parsing and games, service-oriented architectures and cloud computing.

The goal of COMP 405/505 is two-fold: one is to introduce new object-oriented design concepts and the other is raise the level at which those topics are discussed. Students will learn to discuss the OOP/D concepts in terms design patterns, applicability, pros/cons and other programming paradigms. Learning to articulate one's conceptions verbally and in writing are key goals of the course as well as the ability to critically evaluate the opinions expressed by others.

COMP 405 is for undergraduate students and COMP505 is for graduate students. COMP 505 students will have additional responsibilities and requirements of leading the class in the discussions on the various course topics, including but not limited to the preparation of presentation materials and written documentations. COMP 405 students can receive extra credit if they choose to take on the additional required COMP 505 work.

Prerequisites: Comp310 (undergraduates) or Comp504 (graduate students) -- Detailed knowledge and practice in abstract structure and behavioral representations, delegation model programming, design patterns and Java are required.

Primary target audience: Third year or greater CS students who are interested in object-oriented programming and design, large flexible systems, enterprise-class networked applications, dynamically configurable systems and software engineering. Students who plan on taking Comp410 in the future must take Comp310 or Comp404 beforehand and are encouraged to take Comp405/505 as well in preparation for Comp410.

For more and the latest information, please see the Info page or contact Dr. Wong (swong at rice.edu)

CLASS TIMES: MWF 10:00 AM - 10:050 PM

Tentative Schedule

This schedule should only be considered as a guide. Expect that the schedule will change often!

Instructions on how to use Subclipse/Subversion to turn-in your assignments.

General Assignment Instructions (supplemented by the specific instructions linked below)

Date	Lecture	Notes	Assignment Due Date (at beginning of class!)
Week 1	Immutable and Mutable Recursive Structures	• List Structure and the Composite Design Pattern • Visitor Design Pattern • State Design Pattern	
		• Mutable Linear Recursive Structure (Click "**`Legacy Site`**" for proper formatting) ○ Some LRS exercises (demo code in SVN repo)	
		• Binary Tree Structure (Click "**`Legacy Site`**" for proper formatting) ○ Tournament Simulation • State Pattern in a Calculator	
Week 2	*MLK Holiday (typically) -- No Class!*		
	Lazy Evaluation		Finite State Machine
	Additonal lazy evaluation information	No more `if`'s! See lazy evaluation demo code.	
Week 3		Self-Modifying Finite State Machine Demo	

	Design Patterns for Sorting		
Week 4	Extended Visitors and Recursive Descent Parsing	• PPT lecture #1 • PPT lecture #2 -- concentrate on the theoretical parts at the beginning since our implementation will be different, though with the same basic idea. • Self-paced lab -- based on the implementation in lecture #2 above. • Dr. Nguyen and Dr. Wong's published paper and presentation	
			(*New due date!*) Extended Finite State Machine
Week 5			
			(*New due date!*) Operational parser-generator infrastructure
Week 6			HW03 draft write-ups signed off by a staff person
Week 7		Dynamic communications test day	

		Dynamic communications demo day	Dynamic Communications Language
Week 8 (typically)	*Spring Break -- No Classes!*		
Week 9	Object-Oriented Game Theory	System Block Diagrams	
Week 10			HW04 draft write-ups signed-off by staff person
			HW03 peer reviews due
	Software Development	Games demo day	Game Theory
Week 11	Google App Engine		
			Operational GAE project using multiple JSPs and a simple form (everyone)
Week 12			
			Final Project Proposals Due
(typically)	*Midterm Recess -- No Class!*		
Week 13			

Week 14			
Week 15			
TBA (Regular scheduled final exam period)	Final Project Demo Day		Final Project: Cloud Games

---

© 2016 by Stephen Wong

Course Information

TIME AND LOCATION:

Class will be held every Monday, Wednesday and Friday from 10:00 AM - 10:50 AM

INSTRUCTORS:

Dr. Stephen Wong -- See home page for contact and office hour information.

Teaching Assistants	**e-mail (at rice.edu)**	**Office Hours & Location (Subject to change, so check often!)**
TBA		

All Comp405 staff are also available by appointment!

E-Mail:

Entire class (including staff): comp405sXX at owlspace-ccm.rice.edu (where XX is the year and replace the " at " with "@")

Please do not send e-mail directly to the instructor or any specific staff member, as doing so will only delay the response.

> ***IMPORTANT**: In order to send a message to the above address, the e-mail address in your Owlspace address MUST MATCH YOUR RETURN ADDRESS EXACTLY. Generally, this means that you must send from that same account but even that can fail if you use a different alias as the return address.*

Just the staff: comp405 at rice.edu (replace the " at " with "@")

Please DO start your subject line with "**Comp405:** " so that your message can be easily identified as being class-related!

TEXT BOOKS:

Though none of the existing text books truly meet our needs and requirements, we like the following book and highly recommend it for the course.

- Head First Design Patterns, First Edition, by Eric Freeman and Elizabeth Freeman, O'Reilly, 2004. ISBN 0-596-00712-4.

We also have a few other recommendations.

- The Java Programming Language, 4th edition, by Arnold, Gosling, and Holmes, Addison-Wesley,
  James Gosling is the creator of Java. This is the definitive Java reference. However, this book does not cover all the topics of the course and will be used only as a reference for Java syntax and semantics.

- Design Patterns, Elements Of Reusable Object-Oriented Software, by Gamma, Helm, Johnson, and Vlissides, Addison-Wesley, 1995. ISBN 0201633612
  This is the bible of the pattern community. It predates Java. The authors are affectionately referred to as the Gang of Four (GoF). All examples are written in C++ and/or Smalltalk. It can be hard to read if you are not familiar with these two languages, though the ideas are language independent. The "pattern language" used in this book also requires a lot of getting used to. We have learned so much from this book, and we are still reading it and learning from it. There is a version on CD-ROM which is cheaper (and perhaps, more versatile). We strongly recommend it, especially to any student considering further work in object-oriented programming.

- Thinking in Java, 4'th ed., by Bruce Eckel, Prentice Hall PTR.
  Good OO treatment, with a little more detail on syntax than Gosling's book. Students often find it useful for "nuts-and-bolts" information on programming. The full text of earlier editions are available as a free download from www.bruceeckel.com. Also at the web site is his on-line only book: Thinking in Patterns with Java, which is recommended reading.

COURSE PREREQUISITE:

COMP 310

COMP 504

COURSE DESCRIPTION:

COMP 405/505 is a topics-driven exploration of cutting-edge object oriented design issues and concepts including mutable recursive data frameworks, design patterns for lazy evaluation, sorting, parsing and games, service-oriented architectures and cloud computing. Detailed knowledge and practice in abstract structure and behavioral representations, delegation model programming, design patterns and Java are required.

Tentative Topic List

(in no particular order, not all topics may be covered due to time constraints)

- State Pattern

- Lazy Evaluation
- Parsing
- Mutable list frameworks
- Sorting frameworks
- Agent simulations
- Service-oriented architectures
- Cloud computing
- Game theory
- Tournament
- Map-reduce
- Aspect oriented programming

COURSE SCHEDULE:

Please follow the navigation link to the Home page above to see the topical schedule for lectures and labs.

Homework turn-in:

The turn-in mechanism for code will be the student's individual Comp405 Subversion repository on CLE@R, using the Subclipse version control plug-in in Eclipse. Other work will be turned in via postings on the course wiki. Please see

- Instructions on how to use Subclipse/Subversion to turn-in your assignements.
- General instuctions on using the Subclipse plug-in for Eclipse
- General Assignment Guidelines

GRADING:

The course grade will be based on: *TENTATIVE*

COMP 405	COMP 505
• Homework Assignments - 65% • Peer Reviews -- 15% • Final Project -- 20%	• Homework Assignments - 50% • Peer Reviews -- 15% • Teaching – 15% • Final Project -- 20%

Assignments' due dates and rules will be clearly stated at the time they are given.

No make-up exams will be given unless there is a legitimate excuse such as proof of medical emergency. All permissions must be requested and approved in writing (hard copy or e-mail) BEFORE the due date!

We reserve the right to modify the above weightings at any time.

LATE POLICY:

- No assignments will accepted late without express written permission from an instructor **BEFORE** the due date of the assignment. Excused lateness will not be penalized unless otherwise decided by the instructor (student will be notified at time of request).
- Extension requests will be considered ONLY for the following reasons and are NOT automatically accepted:
  - Illness or unexpected travel occurring during or extending into the second half of the time period between the previous assignment due date and the current assignment due date.
  - Pre-planned travel that covers more than half of the time period between the previous assignment due date and the current assignment due date.
  - Unexpected technical problems that prevent the submission of the assignment such as a network or SVN outage.
    - Note that broken computers or faulty software installations are not acceptable justifications because all work can be completed and submitted from the OEDK lab computers.
- The following will NOT be considered as valid reasons for extensions:
  - Excessive work in other classes.
  - Travel to conferences where one is gone for less than half of the assignment time period. This includes the preparation for such trips.
  - Starting late and/or discovering that the assignment is more difficult than anticipated.
  - Not being able to meet with an assignment partner, if one is required by the assignment.
- Any extension request must be accompanied by proof of having already made a concerted effort to complete the assignment, including SVN commits and meetings with the staff. Extensions will not be granted if no work has been done already on the assignment.
- ***If a late, unexcused submission is accepted at all, which is solely at the instructor's discretion and which is NOT the default policy:***
  - The assignment will be penalized *at least* 5% of the total possible score per day on top of any normal markdowns, rounded *up* to the nearest day.
  - If more one week after the due date, the assignment will not be accepted and will be given an automatic "0" score.

Technical and Development Resources:

- **Comp310 Java Resources**
- SVN Turn-in Instructions (See Subclipse plug-in instructions above for how to do this from inside of Eclipse, using the information at this link.)
- **Using Eclipse**
  - Eclipse and plugin-in installation
  - Getting started with Eclipse
  - Using the Subclipse Subversion plug-in
- Dr. Wong and Dr. Nguyen's research papers on OOP/D: http://www.bandgap.cs.rice.edu/personal/adrice_swong/public/WebPages/research/default.htm

Related Course Materials:

Principles of Object Oriented Programming courses:

- Connexions modules: http://cnx.org/content/col10213/latest/
- Comp201: http://www.clear.rice.edu/comp201
- Comp202: http://www.clear.rice.edu/comp202
- Comp211: https://wiki.rice.edu/confluence/display/cswiki/s11
- Comp215: http://www.clear.rice.edu/comp215
- Comp310: http://www.clear.rice.edu/comp310

HONOR SYSTEM:

Rice University has a student-run academic honor system. Each student has personal responsibility for honesty in their own academic conduct. Each student has the responsibility to report any suspected problems to the Honor Council in a timely fashion. During any Rice exam, the student is expected to write the honor pledge on the outside of the exam:

On my honor, I have neither given nor received aid on this examination.

The Computer Science Department takes the Rice Honor System very seriously and handles all violations accordingly. General rules of conduct concerning collaboration will be spelled out before assignments or exams are distributed (see for instance, above). When in doubt, contact an instructor before proceeding.

For more information concerning the Rice Honor System, see:

- The Honor Council home page
- The Constitution of the Honor System
- Recommendations for conducting exams

ADA:

Any student with a documented disability needing academic adjustments or accommodations is requested to speak with us during the first two weeks of class. All discussions will remain confidential. Students with disabilities should also contact Disability Support Services in the Ley Student Center.

© 2016 by Stephen Wong